CS1231 Reference Notes

github.com/reidenong/cheatsheets, AY23/24 S1

Chapter 1. Propositional Logic

Rules of Inference:

Modus Ponens	$p \to q$ p $\therefore q$
Modus Tollens	$p \to q$ $\sim q$ $\therefore \sim p$
Generalization	p $\therefore p \vee q$
Specialization	$p \wedge q$ $\therefore p$
Conjunction	p q $\therefore p \wedge q$
Elimination	$p \vee q$ $\sim p$ $\therefore q$
Transitivity	$p \to q$ $q \to r$ $\therefore p \to r$
Proof by Division into Cases	$p \vee q$ $p \to r$ $q \to r$ $\therefore r$
Contradiction	$\sim p \to F$ $\therefore p$

Chapter 3. Quantified Statements

Expressions of if-then statements:

$r(x)$ sufficient condition for $s(x)$	$r(x) \to s(x)$
$r(x)$ necessary condition for $s(x)$	$\neg r(x) \to \neg s(x)$ $s(x) \to r(x)$
$r(x)$ only if $s(x)$	$r(x) \to s(x)$

Truth Set:
If $P(x)$ is a predicate and x has domain D, the truth set of $P(x)$ is the set of all elements in D that make $P(x)$ true.

Universal Statements:
$\forall x \in D, Q(x)$ where D is the domain of x and $Q(x)$ is a predicate.

Existential Statements:
$\exists x \in D, Q(x)$ where D is the domain of x and $Q(x)$ is a predicate.

Validity of Arguments:
An argument is valid if and only if the truth of its premises implies the truth of its conclusion.

Soundness of Arguments:
An argument is sound if and only if it is valid and all its premises are true.

Showing Invalidity of Arguments:

1. Show there exist a counterexample of predicates such that all premises are true but the conclusion is false.
   - May be easier to work backwards, ie. find how the $\therefore$ can be false before constructing premises
2. Boolean algebra

Rule of Inference for Quantified Statements:

Universal Instantiation	$\forall x \in D, P(x)$ $\therefore P(c)$ if $c \in D$
Universal Generalization	$P(c)$ for every $c \in D$ $\therefore \forall x \in D, P(x)$
Existential Instantiation	$\exists x \in D, P(x)$ $\therefore P(c)$ for some $c \in D$
Existential Generalization	$P(c)$ for some $c \in D$ $\therefore \exists x \in D, P(x)$

Proving Existential Statements
$\exists x \in D, P(x)$:
1. Constructive proof to find a $x \in D$
2. Constructive proof by giving directions to find such an $x \in D$

Disproving Universal Conditional Statement
$\forall x \in D, P(x) \to Q(x)$:
1. Show negation is true by counterexample, ie. prove $\exists x \in D, P(x) \wedge \sim Q(x)$

Chapter 5. Sets

Set Builder Notation:
The set of all x such that $P(x)$ is true is denoted by

$$A = \{x \in D \mid P(x)\}$$

where D is the domain of x and $P(x)$ is a predicate.

Set Replacement Notation:
The set of all x such that $P(x)$ is true is denoted by

$$A = \{f(x) \mid x \in D\}$$

where D is the domain of x and $f(x)$ is a function.

Roster Notation:
Listing all elements, we have

$$A = \{a, b, c, ...\}$$

Disjoint Sets:
Two sets A and B are disjoint if and only if they have no elements in common, ie. $A \cap B = \emptyset$.

Power Sets
The power set of a set $P(A)$ is the set of all subsets of A. For a set A with n elements, $P(A)$ has 2^n (Theorem 5.2.4)

For all sets A and B,

$$P(A \cap B) = P(A) \cap P(B)$$

However, there exists A and B such that

$$P(A \cup B) \neq P(A) \cup P(B)$$

Proving with Sets
1. Work with the universal set U, convert to boolean algebra and then back to sets
   eg. let $z \in U$, then $z \in$ {Given Set} ...
2. Work with set notation

Chapter 6. Relations

Definition of a relation:
A relation R from a set A to a set B is a subset of the Cartesian product AxB.

Given statement P(x,y), we have

$$\forall (x, y) \in A \times B, ((x, y) \in R \leftrightarrow P(x, y))$$
$$\forall x \in A, \forall y \in B, (xRy \leftrightarrow P(x, y))$$

Inverse Relations:
Given a relation R from a set A to a set B, the inverse relation $R^{\{-1\}}$ from B to A is defined as

$$R^{-1} = \{(y, x) \in B \times A : (x, y) \in R\}$$
$$\forall x \in A, \forall y \in B, (xRy \leftrightarrow yR^{-1}x)$$

Domain, Co-Domain and Range:
Domain of R is the set of all first elements of ordered pairs in R, ie.

$$\{x \in A : \exists y \in B, (x, y) \in R\}$$

Co-domain of R is the set of all second elements of ordered pairs in R, ie. B

Range of R is the set of all second elements of ordered pairs in R, ie.

$$\{y \in B : \exists x \in A, (x, y) \in R\}$$

Compositions of Relations:
Relation starting in R and ending in S
$=$ Composition of R with S
$= S \circ R$

$$\forall x \in A, \forall z \in C, (x(\boldsymbol{S} \circ \boldsymbol{R})z \leftrightarrow \exists y \in B, (xRy \wedge ySz))$$

Composition is associative, ie.

$$(S \circ R) \circ T = S \circ (R \circ T) = S \circ R \circ T$$

Inverse of Composition is given as

$$(S \circ R)^{-1} = R^{-1} \circ S^{-1}$$

Properties of Relations

Reflexive	$\forall x \in A, xRx$
Irreflexive	$\forall x \in A, (x, x) \notin R$
Symmetric	$\forall x \in A, \forall y \in A, (xRy \to yRx)$
Anti-Sym	$\forall x \in A, \forall y \in A, ((xRy \wedge yRx) \to x = y)$
Asymmetric	$\forall x \in A, \forall y \in A, (xRy \to y\not{R}x)$ ie. Anti-Sym and Irreflexive
Transitive	$\forall x, y, z \in A, ((xRy \wedge yRz) \to xRz)$
Equivalence	Reflexive, Symmetric, Transitive
Partial Order	Reflexive, Anti-Sym, Transitive

Transitive Closure:
The transitive closure of a relation R on a set A is the smallest transitive relation on A that contains R.

Partitions
A partition of a set A is a collection of non-empty, mutually disjoint subsets of A such that every element of A is in exactly one of these subsets.

λ is a partiton of set A if
1. λ is a set of non-empty subsets of A
2. Every element of A is in exactly one element of λ, ie.

$$\forall x \in A, \exists S \in \lambda(x \in S)$$
$$\forall x \in A, \forall S, T \in \lambda((x \in S \wedge x \in T) \to S = T)$$

Equivalence Relations
Relations induced by set partitions are equivalence relations.

The set of all elements $x \in A$ such that A is ~-related to x is known as the *equivalence class* of x and is denoted by $[x]$.

$$[a] = \{x \in A : x \sim a\}$$
$$\forall x \in A, \forall y \in A, ([x] = [y] \leftrightarrow x \sim y)$$

Order Relations

Maximal Element	c is a maximal element iff $$\forall x \in A, c \preccurlyeq x \to c = x$$ ie. no larger element exists
Largest / Greatest/ Maximum Element	c is a largest element iff $$\forall x \in A, x \preccurlyeq c$$ ie. all other elements are smaller

Minimal Element	c is a minimal element iff $$\forall x \in A, x \preccurlyeq c \rightarrow c = x$$ ie. no smaller element exists
Smallest / Least / Minimum Element	c is a smallest element iff $$\forall x \in A, c \preccurlyeq x$$ ie. all other elements are larger

Compatible and Comparable

Consider a partial order $\preccurlyeq$ on a set A, with $a, b \in A$.

$$\text{a, b comparable} \leftrightarrow a \preccurlyeq b \vee b \preccurlyeq a$$
$$\text{a, b compatible} \leftrightarrow \exists c \in A, (a \preccurlyeq c \wedge b \preccurlyeq c)$$

Definition of a Total Order:

$$\forall x, y \in A, (xRy \vee yRx)$$

Chapter 7. Functions

Definition of a function:

A function f from a set A to a set B, $f : X \rightarrow Y$, is a relation from A to B such that

$$\text{(F1) } \forall x \in X, \exists y \in Y, (x, y) \in f$$

$$\text{(F2) } \forall x \in X, \forall y_1 \in Y, \forall y_2 \in Y,$$
$$((x, y_1) \in f \wedge (x, y_2) \in f \rightarrow y_1 = y_2)$$

In other words,

$$\forall x \in A, \exists! y \in B, (x, y) \in f$$

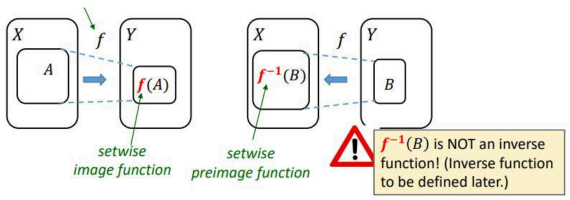
Terminology:

(1) If $f : X \rightarrow Y$ and $(x, y) \in f$, then x is the pre-image of y and y is the image of x.
(2) The domain of f is X and the co-domain of f is Y.
(3) The range of f is the set of all images of elements in X, and Range $\subseteq$ Co-domain.

Setwise Image and Pre-Image:

Given a function $f : X \rightarrow Y$, and $A \subseteq X$ and $B \subseteq Y$ are some sets,

$$f[A] = \{f(x) : x \in A\} \text{ is the image of subset A and}$$
$$f^{\{-1\}}[B] = \{x \in X : f(x) \in B\} \text{ is the preimage of subset B}$$



Two functions $f : A \rightarrow B$ and $g : C \rightarrow D$ are equal if and only if $A = C$, $B = D$ and $f(x) = g(x)$ for all $x \in A$.

Injective, Surjective and Bijective Functions

A function $f : X \rightarrow Y$ is *injective* if and only if

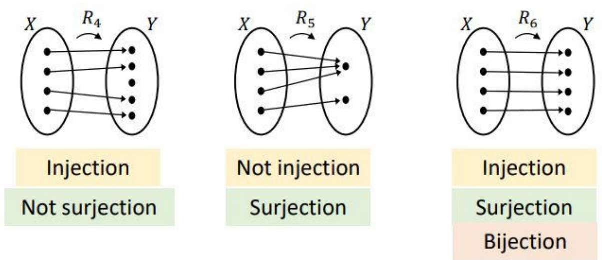
$$\forall x_1, x_2 \in X, (f(x_1) = f(x_2) \rightarrow x_1 = x_2)$$

A function $f : X \rightarrow Y$ is *surjective* if and only if

$$\forall y \in Y, \exists x \in X, (f(x) = y)$$

A function $f : X \rightarrow Y$ is *bijective* if and only if it is both injective and surjective, ie.

$$\forall y \in Y, \exists! x \in X, (f(x) = y)$$



Order of a bijection

The *order* of a bijection $f : X \rightarrow X$ is the smallest positive integer n such that $f^n = f \circ f \circ \ldots \circ f$ (n times) is the identity function on X.

Inverse Functions

A function $f : X \rightarrow Y$ has an inverse function $f^{-1} : Y \rightarrow X$ if and only if f is bijective.

$$\forall x \in X, \forall y \in Y, (f(x) = y \leftrightarrow f^{-1}(y) = x)$$

Left and Right Inverses

Given a function $f : A \rightarrow B$,

1. $g : B \rightarrow A$ is a left inverse of f if and only if $g \circ f = g(f(a)) = a$ for all $a \in A$. A left inverse exists if and only if f is injective.
2. $h : B \rightarrow A$ is a right inverse of f if and only if $f \circ h = f(h(b)) = b$ for all $b \in B$. A right inverse exists if and only if f is surjective.

Composition of Functions

Given functions $f : X \rightarrow Y$ and $g : Y \rightarrow Z$, the composition of g with f is the function $(g \circ f) : X \rightarrow Z$ defined by

$$\forall x \in X, (g \circ f)(x) = g(f(x))$$

Properties of Compositions:

1. Associative: $(h \circ g) \circ f = h \circ (g \circ f)$
2. Not Commutative: $g \circ f \neq f \circ g$
3. If f and g are both injective, then $g \circ f$ is injective
4. If f and g are both surjective, then $g \circ f$ is surjective

Sequences

A sequence $a_0, a_1, a_2, \ldots$ can be represented by a function a whose domain is $\mathbb{Z}_{\geq 0}$ that satisfies $a(n) = a_n$ for all $n \in \mathbb{Z}_{\geq 0}$.

Fibonacci is a sequence where $a_0 = 0, a_1 = 1$ and $F_{n+2} = F_{n+1} + F_n$ for all $n \in \mathbb{Z}_{\geq 0}$.

Strings

let A be a set. A string over A is an expression of the form $a_0 a_1 a_2 \ldots a_{l-1}$ where $l \in \mathbb{Z}_{\geq 0}$ and $a_0, a_1, \ldots \in A$. l is the length of the string, and the empty string ε is the string of length 0.

Sequence/String Equality

Two sequences $a_0, a_1, a_2, ...$ and $b_0, b_1, b_2, ...$ defined respectively by functions $a(n) = a_n$ and $b(n) = b_n$ are equal if and only if $a(n) = b(n)$ for all $n \in \mathbb{Z}_{\geq 0}$.

Two strings $a_0 a_1 a_2 ... a_{l-1}$ and $b_0 b_1 b_2 ... b_{m-1}$ are equal if and only if $l = m$ and $a_i = b_i$ for all $i \in \mathbb{Z}_{\geq 0}$.

Addition and Multiplication on $\mathbb{Z}_n$

Given $a, b \in \mathbb{Z}$ and $n \in \mathbb{Z}_{>0}$, $a \equiv b (\operatorname{mod} n)$ if and only if n divides $a - b$, ie. $a - b = nk$, for some $k \in \mathbb{Z}$.

$\mathbb{Z}_n$ is the set of all equivalence classes of $\mathbb{Z}$ under congruence modulo n, eg.

$$\mathbb{Z}_2 = \{\{2k : k \in \mathbb{Z}\}, \{2k + 1 : k \in \mathbb{Z}\}\},$$
$$\mathbb{Z}_3 = \{\{3k : k \in \mathbb{Z}\}, \{3k + 1 : k \in \mathbb{Z}\}, \{3k + 2 : k \in \mathbb{Z}\}\}$$

Whenever $[a], [b] \in \mathbb{Z}_n$, addition and multiplication on $\mathbb{Z}_n$ is defined as

$$[a] + [b] = [a + b]$$
$$[a] * [b] = [a * b]$$

Addition and multiplication on $\mathbb{Z}_n$ is well-defined, meaning it does not depend on the choice of representatives of the equivalence classes. ie.
$\forall n \in \mathbb{Z}^+$ and all $[x_1], [y_1], [x_2], [y_2] \in \mathbb{Z}_n$,

$$[x_1] = [x_2] \wedge [y_1] = [y_2] \to [x_1] + [y_1] = [x_2] + [y_2]$$
$$[x_1] = [x_2] \wedge [y_1] = [y_2] \to [x_1] \cdot [y_1] = [x_2] \cdot [y_2]$$

Well Defined Properties

In general, a *function* $f : X \to Y$ is well-defined if and only if

$$\forall x_1, x_2 \in X, (x_1 = x_2 \to f(x_1) = f(x_2))$$

With respect to an *equivalence relation* ~, f is well defined if and only if

$$\forall x_1, x_2 \in X, \forall f : X \to Y (x_1 \sim x_2 \to f(x_1) = f(x_2))$$

With respect to an *equivalence class* $[x]$, f is well defined if and only if

$$\forall x_1, x_2 \in X, \forall f : X \to Y ([x_1] = [x_2] \to [f(x_1)] = [f(x_2)])$$

Chapter 8. Mathematical Induction

Sums of sequences

Suppose we have a arithmetic sequence $a_n = a_0 + d \cdot n$, then the sum of the first n terms is given by

$$\sum_{k=0}^{n-1} = \left(\frac{n}{2}\right)(2a_0 + (n-1)d)$$

Suppose we have a geometric sequence $a_n = a_0 \cdot r^n$, then the sum of the first n terms is given by

$$\sum_{k=0}^{n-1} = a_0 \frac{1 - r^n}{1 - r}$$

Principle of Mathematical Induction

1PI / Weak Induction:

(1) $P(a)$ is true
(2) $\forall k \geq a, P(k) \to P(k + 1)$
$(\therefore)$ $\forall n \geq a, P(n)$

2PI / Strong Induction:

(1) $P(a)$ is true
(2) $\forall k \geq a, (P(a) \wedge ... \wedge P(k)) \to P(k + 1)$
$(\therefore)$ $\forall n \geq a, P(n)$

(1) $P(a), P(a + 1), ..., P(b)$ is true
(2) $\forall k \geq a, P(k) \to P(k + b - a + 1)$
$(\therefore)$ $\forall n \geq a, P(n)$

Well Ordering Principle

The well ordering principle for the Integers states every non-empty subset of $\mathbb{Z}_{\geq 0}$ has a least element.
The well ordering principle for non-negative integers state every non-empty subset of $\mathbb{Z}_{\geq 0}$ has a least element.

Recurrence Relations

Recursive definition of a set S

<u>Base clause</u>
Specify that certain elements (founders) are in S.
ie. if c is a founder, then $c \in S$
<u>Recursive clause</u>
Specify that if certain elements are in S, then certain other elements are in S.
$\to$ given constructor f, if $x \in s, f(x) \in s$.
<u>Minimality clause</u>
Membership for S can be demonstrated by application of the above clauses

Structural Induction

In proving that $\forall x \in H, P(n)$ is true,
<u>Basis</u>
Show that P(c) is true for **all** founders c
<u>Induction</u>
Show that $\forall x \in S, P(x) \to P(f(x))$ is true for all constructors f.

Chapter 9. Cardinality

Pigeonhole Principle

Let A and B be finite sets. If there is a injection $f : A \to B$, then $|A| \leq |B|$. By the contrapositive, with $m, n \in \mathbb{Z}^+, m > n$, if m pigeons are put into n pigeonholes, there must be at least one pigeonhole with more than one pigeon.

Let A and B be finite sets. If there is a surjection $f : A \to B$, then $|A| \geq |B|$. By the contrapositive, with $m, n \in \mathbb{Z}^+, m < n$, if m pigeons are put into n pigeonholes, there must be at least one pigeonhole with no pigeon.

Application of PHP to Decimal expansions of Fractions:
Considering $\frac{a}{b}$, let $r_0 = a$ and $r_1, r_2, \ldots$ be the remainders obtained in the long division of a by b.
By the quotient-remainder theorem, each remainder r_i is an integer between 0 and $b-1$.
if remainder $r_i = 0$, then the division terminates, else it goes on indefinitely. By the PHP, since there ar emore remainders than values that remainders can take, some remainder value must repeat $r_j = r_k$. It follows that the decimal expansion of $\frac{a}{b}$ is periodic, with period $k-j$.

Definitions of Cardinality
Two sets A and B have the same cardinality if and only if there exists a bijection $f : A \to B$.

A set A is finite if and only if $A = \emptyset$, or there exists a bijection from S to $\mathbb{Z}_n$ for some $n \in \mathbb{Z}^+$.

A set A is countably infinite if and only if there exists a bijection from A to $\mathbb{Z}_{\geq 0}$, ie. $|A| = |\mathbb{Z}^+| = \aleph_0$

Countability of a set
A set is countable iff it is finite or countably infinite. A set is uncountable if it is not countable.

If sets A and B are both countable, then $A \times B$ is countable. (Theorem 9.2.5) This extends to $A_1 \times A_2 \times \ldots \times A_n$ and $A_1 \cup A_2 \cup \ldots$ (Union to infinity).

An infinite set is countable iff there is a sequence $b_0, b_1, \ldots \in B$ in which every element of B appears exactly once (Proposition 9.1). As a result, an infinite set B is countable iff there is a sequence $b_0, b_1, \ldots$ in which every element of B appears (Lemma 9.2).

Larger Infinities
The set of real numbers between 0 and 1, $(0,1) = \{x \in R \mid 0 < x < 1\}$ is uncountable. (Theorem 7.4.2). To prove a set is uncountable means proving that there is no possibility of a bijection from that set to $\mathbb{Z}^+$.

Any subset of any countable set is countable (Theorem 7.4.3), and any set with an uncountable subset is uncountable (Corollary 7.4.4).

Every infinite set has a countably infinite subset (Proposition 9.3)

Cantor's Diagonalization Argument
1. Suppose that $(0,1)$ is countable.
2. Since it is not finite, it is countably infinite.
3. We list the elements x_i of $(0,1)$ in a sequence.

$$\begin{aligned} x_1 &= 0.a_{11}a_{12}a_{13}\ldots a_{1n}\ldots \\ x_2 &= 0.a_{21}a_{22}a_{23}\ldots a_{2n}\ldots \\ &\vdots \\ x_n &= 0.a_{n1}a_{n2}a_{n3}\ldots a_{nn}\ldots \\ &\vdots \end{aligned}$$

4. We construct a number $d = 0.d_1d_2\ldots d_n\ldots$ such that
$$\text{d} = \begin{cases} 1, \text{if } a_{nn} \neq 1; \\ 2, \text{if } a_{nn} = 1. \end{cases}$$
5. $\forall n \in \mathbb{Z}^+, d_n \neq a_{nn}$. Thus, $d \neq x_n, \forall n \in \mathbb{Z}^+$.
6. But clearly $d \in (0,1)$, hence a contradiction. $\therefore$ (0,1) is uncountable.

Chapter 10: Counting and Probability

Binomial Theorem:

$$(a+b)^n = \sum_{k=0}^{n} \binom{n}{k} a^{n-k} b^k$$

Some common identities as a result are
1. $\sum_{k=0}^{n} \binom{n}{k} = 2^n$
2. $\sum_{k=0}^{n} (-1)^k \binom{n}{k} = 0$
3. $\sum_{k=0}^{n} 2^k \binom{n}{k} = 3^n$

nCk, nPk Formula:

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}, \quad \text{P}(n,k) = \frac{n!}{(n-k)!}$$

PIE:
For finite sets A, B and C,

$$|A \cup B| = |A| + |B| - |A \cap B|$$

$$\begin{aligned} |A \cup B \cup C| = |A| + |B| + |C| - |A \cap B| \\ -|A \cap C| - |B \cap C| + |A \cap B \cap C| \end{aligned}$$

r-Combinations with repetitions

$$\text{ways } (x_1 + x_2 + \ldots + x_n = r) = \binom{n+r-1}{r}$$

Pascal's Formula
For positive integers n and r, $r \leq n$,

$$\binom{n+1}{r} = \binom{n}{r-1} + \binom{n}{r}$$

Methods of combinatorial Proofs
1. Algebraic Proofs
2. Bijective Proofs
3. Double Counting Proofs

Probability
Probability of a General Union of events:

$$P(A \cup B) = P(A) + P(B) - P(A \cap B)$$

Conditional Probability:

$$P(A \mid B) = \frac{P(A \cap B)}{P(B)} \quad \text{(Theorem 9.9.1)}$$

$$P(A \cap B) = P(A \mid B)P(B) \quad \text{(Theorem 9.9.2)}$$

$$P(A) = \frac{P(A \cap B)}{P(B \mid A)} \quad \text{(Theorem 9.9.3)}$$

Bayes' Theorem General form:
Suppose a sample space S is a union of mutually disjoint events $B_1, B_2, \ldots, B_n$, A is an event in S, and A and all B_i have non zero probailities. Then

$$P(B_k \mid A) = \frac{P(A \mid B_k) \cdot P(B_k)}{\sum_{i=1}^{n} P(A \mid B_i) P(B_i)}$$

Independent Events
Events A and B are independent if and only if
$P(A \mid B) = P(A)$ or $P(B \mid A) = P(B)$,
ie. $P(A \cap B) = P(A) \cdot P(B)$

Pairwise independence is when 3 events A, B, C are such that $P(X \cap Y) = P(X) \cdot P(Y)$, for X,Y in {A,B,C}.

Mutual independence is when 3 pairwise independent events are such that
$P(A \cap B \cap C) = P(A) \cdot P(B) \cdot P(C)$.

Chapter 11: Graphs

Simple Graph
An undirected graph with no loops or parallel edges.

Complete Graph K_n
A simple graph with n vertices and exactly one edge between every pair of vertices. There are a total of $\binom{n}{2} = \frac{n(n-1)}{2}$ edges.

Bipartite Graph
A simple graph whos vertices can be divided into two disjoint sets U and V such that every edge connects a vertex in U to one in V.
A complete bipartite graph $K_{m,n}$ is where every vertex in U connects to every vertex in V.

Subgraph
H is a subgraph of G iff every vertex in H is a vertex in G and every edge in H is an edge in G, and every edge in H has the same endpoints as it did in G.

Degree of a Vertex
The number of edges incident to a vertex (loops count twice). The total degree of the graph is the sum of the degrees of all the vertices.

The handshake theorem (Theorem 10.1.1) states

$$\text{total degree of G} = \deg(v_1) + \ldots + \deg(v_n) = 2|E|$$

.

Then it follows that the total degree of a graph is even (Corollary 10.1.2), and in any graph there are an even number of vertices of odd degree (Proposition 10.1.3).

In a directed graph,

$$\sum_{v \in V} \deg^+(v) = \sum_{v \in V} \deg^-(v) = |E|$$

Walks, Paths and Circuits
Walk
A walk in a graph G is a finite alternating sequence of vertices and edges of the form $v_0, e_1, v_1, e_2, v_2, \ldots, e_n, v_n$.
A closed walk is a walk that starts and ends at the same vertex.

A circuit is a *cycle* of at least length 3 with no repeated edges. A simple circuit is a circuit with no repeated vertices except the first and last.

In a simple, undirected graph, if there are two distinct paths from u to v, then there is a circuit containing both paths. (Lemma 10.5.5)

Trails and Paths
A trails is a walk with no repeated edges. A path is a trail with no repeated vertices.

Connected Graphs
Two vertices are connected (reachable) iff and only if there is a walk from one to the other. The graph is connected if and only if every pair of vertices is connected. All connected graphs have a spanning tree (Proposition 10.7.1)

Connected Components
A connected component of a graph is a connected subgraph of the largest possible size.

A subgraph of a directed graph is considered to be a Strongly Connected Component iff for every pair of vertices A and B, there exists a path from A to B and a path from B to A.

Eulerian Circuits
A euler circuit of G contains every vertex and traverses every edge of G exactly once.

$$\text{G has a euler circuit} \longrightarrow \\ \text{every vertex has even degree} \land \text{G is connected}$$

An euler path from u to v is a path that traverses every vertex at least once, every edge of G exactly.

$$\text{G has a euler path} \longrightarrow \text{ONLY u and v have odd degree}$$

Hamiltonian Circuits
A hamiltonian circuit of G contains every vertex of G exactly once.

Properties of Hamiltonian Circuits:
If G has a Hamiltonian circuit, then it has subgraph H where
1. H contains every vertex of G
2. H is connected
3. H has the same number of edges as vertices
4. Every vertex of H has degree 2

(Proposition 10.2.6)

Isomorphism
G is isomorphic to G' iff there exist bijections

$$g : V(G) \to V(G') \text{ and } h : E(G) \to E(G')$$

that preserve the edge-endpoint functions such that

$$v \text{ is an endpoint of } e \leftrightarrow g(v) \text{ is an endpoint of } h(e)$$

Graph isomorphism is an equivalence relation on a set of graphs, ie. it is reflexive, symmetric and transitive.

Planar Graphs
A graph that can be drawn on a 2D graph without crossing edges.

By Kuratowski's Theorem, a graph is planar iff it does not contain a subgraph that is a subdivision of K_5 or $K_{3,3}$.

Euler's Formula
For a connected planar graph with v vertices, e edges and f faces, $f = e - v + 2$.

Chapter 12. Trees

A graph is a tree if it is circuit-free and connected, and a forest if it is circuit-free and not connected.

Properties of trees:

1. Any non-trivial tree has at least one vertex of degree 1. (Lemma 10.5.1)
2. Any tree with n vertices has $n-1$ edges. (Theorem 10.5.2)
3. If G is a connected graph, C a circuit in G, if one of the edges of C is removed from G, the resultant graph is still connected. (Lemma 10.5.3)
4. If G is a connected graph with n vertices and $n-1$ edges, G is a tree.

Rooted Trees

A rooted tree is a tree with a distinguished root.
The *level* of a vertex is the number of edges along the unique path between it and the root. The *height* of a rooted tree is the maximum level of any vertex in the tree.

Binary Trees

A binary tree is a rooted tree where every parent has at most two children.
A full binary tree is one where every parent has exactly two children.

The left subtree of vertice v is the binary tree whose root is the left child of v.

Properties of binary trees:

1. If T is a full binary tree with k internal vertices, then T has a total of $2k+1$ vertices and $k+1$ terminal vertices. (Theorem 10.6.1)
2. if any binary tree T has height h and t terminal vertices , then

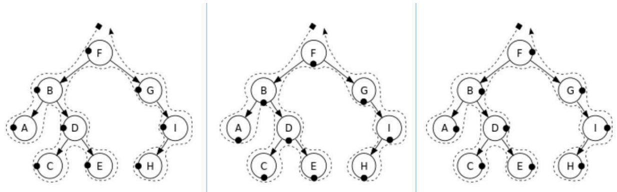
$$t \leq 2^h \leftrightarrow \log_2 t \leq h$$

DFS Variations

Pre-Order
1. Print the root
2. Traverse the left subtree recursively
3. Traverse the right subtree recursively

In-Order
1. Traverse the left subtree recursively
2. Print the root
3. Traverse the right subtree recursively

Post-Order
1. Traverse the left subtree recursively
2. Traverse the right subtree recursively
3. Print the root



Pre-order: F, B, A, D, C, E, G, I, H

In-order: A, B, C, D, E, F, G, H, I

Post-order: A, C, E, D, B, H, I, G, F

Spanning Tree

A spanning tree of a graph G is a subgraph of G that is a tree containing every vertex of G.

Proposition 10.7.1:
1. Every connected graph has a spanning tree
2. Any two spanning trees of a graph G have the same number of edges

The minimum spanning tree for a connected weighted graph is a spanning tree that has the least possible total weight compared to all other spanning trees.

Kruskal's Algorithm

Greedily add all lightest edges to the tree that does not form a cycle, ending when there are $n-1$ edges. For unique output, the array edge sorting by weight must be stable.

```
Input: Connected weighted graph G
Algorithm:
1. T = graph of vertices of G
2. E = set of edges of G
3. while T has less than n-1 edges {
  3a. Find edge e with the least weight
  3b. if (e forms a cycle in T)
        -> delete it
      else
        -> add it to T
      }
```

Prim's Algorithm

Beginning from a single vertex, find a edge with the least weight that connects to a vertex not yet in the tree, and add it to the tree. Repeat until there are $n-1$ edges.
For unique ouput, the array of edge-vertice sorting by weight must be stable.

```
Input: Connected weighted graph G
Algorithm:
1. T = empty graph with one vertex
2. V = set of all other vertices
3. for i in range (1, n-1) {
  3a. Find edge e such that
        (1) e connects T to one vertex in V
        (2) e has the least weight of all
            edges connecting T to a vertex
            in V
  3b. Add e and its vertice in V to T, delete
      it from V.
      }
```

Approaches to Graphs

1. Degree arguments
2. Connected $\rightarrow$ Tree arguments
3. Unconnected $\rightarrow$ k Connected Components
4. Face, edge, vertice arguments

Appendix

Laws of Boolean Algebra:

Commutative Law	$p \wedge q \equiv q \wedge p$	$p \vee q \equiv q \vee p$
Associative Law	$p \wedge (q \wedge r) \equiv (p \wedge q) \wedge r$	$p \vee (q \vee r) \equiv (p \vee q) \vee r$
Distributive Law	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$
Identity Law	$p \wedge T \equiv p$	$p \vee F \equiv p$
Negation Law	$p \wedge {\sim}p \equiv F$	$p \vee {\sim}p \equiv T$
Double Negation Law	${\sim}({\sim}p) \equiv p$	
Idempotent Law	$p \wedge p \equiv p$	$p \vee p \equiv p$
Universal Bound Law	$p \vee T \equiv T$	$p \wedge F \equiv F$
De Morgan's Law	${\sim}(p \wedge q) \equiv {\sim}p \vee {\sim}q$	${\sim}(p \vee q) \equiv {\sim}p \wedge {\sim}q$
Absorption Law	$p \wedge (p \vee q) \equiv p$	$p \vee (p \wedge q) \equiv p$
Negation of T and F	${\sim}T \equiv F$	${\sim}F \equiv T$
Implication Law	$p \to q \equiv {\sim}p \vee q$	
Contrapositive Law	$p \to q \equiv {\sim}q \to {\sim}p$	
Converse Law	$\text{converse}(p \to q) \equiv q \to p$	
Inverse Law	$\text{inverse}(p \to q) \equiv {\sim}p \to {\sim}q$	

Consensus Theorem	$(p \wedge q) \vee (\neg p \wedge r) \vee (q \wedge r) \equiv (p \wedge q) \vee (\neg p \wedge r)$
Proof	$(p \wedge q) \vee \underline{(q \wedge r)} \vee (\neg p \wedge r)$ $\equiv (p \wedge q) \vee \underline{\{(\neg p \vee p) \wedge (q \wedge r)\}} \vee (\neg p \wedge r)$ $\equiv (p \wedge q) \vee (p \wedge q \wedge r) \vee (\neg p \wedge q \wedge r) \vee (\neg p \wedge r)$ $\equiv (p \wedge q) \vee (\neg p \wedge r)$

Laws of Set Algebra

Commutative Law	$A \cup B = B \cup A$	$A \cap B = .B \cap A$
Associative Law	$A \cup (B \cup C) = (A \cup B) \cup C$	$A \cap (B \cap C) = (A \cap B) \cap C$
Distributive Law	$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$	$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$
Identity Law	$A \cup \emptyset = A$	$A \cap U = A$
Complement Law	$A \cup \overline{A} = U$	$A \cap \overline{A} = \emptyset$
Idempotent Law	$A \cup A = A$	$A \cap A = A$
Universal Bound Law	$A \cup U = U$	$A \cap \emptyset = \emptyset$
De Morgan's Law	$\overline{A \cup B} = \overline{A} \cap \overline{B}$	$\overline{A \cap B} = \overline{A} \cup \overline{B}$
Absorption Law	$A \cup (A \cap B) = A$	$A \cap (A \cup B) = A$
Double Complement Law	$\overline{\overline{A}} = A$	
Complement of Universal Set Law	$\overline{U} = \emptyset$	
Set Difference Law	$A \setminus B = A \cap \overline{B}$	

Quick Power Set References

$P(\emptyset)$	$\{\emptyset\} = \{\{\}\}$
$P(\{a\})$	$\{\emptyset, \{a\}\}$
$P(\{a, b\})$	$\{\emptyset, \{a\}, \{b\},$ $\{a, b\}\}$
$P(\{a, b, c\})$	$\{\emptyset, \{a\}, \{b\}, \{c\},$ $\{a, b\}, \{a, c\}, \{b, c\},$ $\{a, b, c\}\}$
$P(\{a, b, c, d\})$	$\{\emptyset, \{a\}, \{b\}, \{c\}, \{d\},$ $\{a, b\}, \{a, c\}, \{a, d\}, \{b, c\}, \{b, d\}, \{c, d\},$ $\{a, b, c\}, \{a, b, d\}, \{a, c, d\}, \{b, c, d\},$ $\{a, b, c, d\}\}$